REPORT DOCUMENTATION PAGE

Form Approved
OMB NO. 0704-0188

1. AGENCY USE ONLY (Leave Blank)

2. REPORT DATE Sept 17th, 2001

3. REPORT TYPE AND DATES COVERED
Final report; July 1st 1998 – June 30th 2001

4. TITLE AND SUBTITLE
Abstraction-based approaches to correct reactive software

5. FUNDING NUMBERS
DAAG55-98-1-0309

6. AUTHOR(S)
S. Purushothaman Iyer

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
North Carolina State University

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

U. S. Army Research Office
P.O. Box 12211
Research Triangle Park, NC 27709-2211

10. SPONSORING / MONITORING AGENCY REPORT NUMBER
38682.1 -C1

11. SUPPLEMENTARY NOTES
The views, opinions and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy or decision, unless so designated by other documentation.

12 a. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release; distribution unlimited.

12 b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

This final reports on our efforts to make abstraction-based reasoning of reactive software possible. Concurrency Workbench, a tool for designs of reactive concurrent systems was enriched to handle programs. The enrichment depends upon notions of abstractions, a technique used in compilers. Furthermore, work was also done on enriching the Concurrency Workbench to deal with probabilistic information regarding errors during a systems execution.

14. SUBJECT TERMS
Concurrent Systems; Reactive Systems; Abstractions; Model-checking; Concurrency Workbench.

15. NUMBER OF PAGES
4

16. PRICE CODE

17. SECURITY CLASSIFICATION OR REPORT
UNCLASSIFIED

18. SECURITY CLASSIFICATION ON THIS PAGE
UNCLASSIFIED

19. SECURITY CLASSIFICATION OF ABSTRACT
UNCLASSIFIED

20. LIMITATION OF ABSTRACT
UL

MASTER COPY: PLEASE KEEP THIS "MEMORANDUM OF TRANSMITTAL" BLANK FOR REPRODUCTION PURPOSES. WHEN REPORTS ARE GENERATED UNDER THE ARO SPONSORSHIP, FORWARD A COMPLETED COPY OF THIS FORM WITH EACH REPORT SHIPMENT TO THE ARO. THIS WILL ASSURE PROPER IDENTIFICATION. NOT TO BE USED FOR INTERIM PROGRESS REPORTS; SEE PAGE 2 FOR INTERIM PROGRESS REPORT INSTRUCTIONS.

MEMORANDUM OF TRANSMITTAL

U.S. Army Research Office
ATTN: AMSRL-RO-BI (TR)
P.O. Box 12211
Research Triangle Park, NC 27709-2211

□ Reprint (Orig + 2 copies)

□ Manuscript (1 copy)

□ Technical Report (Orig + 2 copies)

☒ Final Progress Report (Orig + 2 copies)

□ Related Materials, Abstracts, Theses (1 copy)

CONTRACT/GRANT NUMBER: DAAG55-98-1-0309

REPORT TITLE: Final report on Abstraction-based approaches to correct reactive software

is forwarded for your information.

SUBMITTED FOR PUBLICATION TO (applicable only if report is manuscript):

Sincerely,

Final Report: ARO Project DAAG55-98-1-0309 Abstraction-based approaches to correct Reactive Software

S. Purushothaman Iyer

September 17, 2001

1 Summary of goals

The goal of the research supported by the grant DAAG-55-98-1-0309 from ARO was to increase the ability of the Concurrency Workbench, a tool for specifying and reasoning about reactive systems, to deal with programs, and not just design notations. With this goal in mind we considered the following two main problems:

- Is it possible to build an interface to the Concurrency Workbench such that it accepts C programs as inputs? The motivation for considering this problem is so that the reach of model-checking and simulation relations checking can be extended from finite state designs to programs.
- Can the Concurrency Workbench be extended to handle design notations that have probabilistic components in them?

Furthermore, given that we wish to extend model-checking type techniques to programs, we also considered the problem of simulating concurrent programs.

2 Summary of Important Results

The important results achieved during the project are as follows:

1. C2CCS – a system to abstract C programs based on user input has been constructed. This system, which borrows ideas from work on Compilers and data flow analysis, allows an user to input a C program and instructions on what aspects should be abstracted, and how. For instance, the user could specify that the only meaningful value of a program variable are whether it is positive or negative. With such specifications about all variables in a program the C2CCS converter can build a finite state model of the program that is being analyzed. The output of the converter is a specification in CCS that can be input to the Concurrency workbench, which can then be reasoned

about. As of the writing of the report the tool is almost done, and is being tested on protocol specifications such as the i-Protocol. The principal implementor, Dan DuVarney, is also currently finishing up his dissertation and will defend it very soon.

2. One of the main strengths of Process Algebras, and its implementation in the Concurrency Workbench, is compositionality, i.e., the ability to reason about a system by reasoning about its subsystems. In the presence of probabilistic information, such as the error rate of communication medium, properties can not be established with certainty but with certain probability. We have designed enriched version of the specification language used in the Concurrency Workbench and an enriched version of mu-calculus, the requirements language used in the workbench. This work is ongoing and is supported by a new grant from ARO.

3. With colleagues from France, the PI investigated how abstractions can be used for simulating programs. While simulations can be used to find the presence of bugs and not the absence of bugs, it forms a major component in any program development environment. To extend the reach of our techniques we investigated how symbolic representations could be used to drive the simulation faster, and get it to cover larger parts of a program's execution. The main result is that a class of regular expressions can be effectively used in computing the interactions among a set of processes that use buffered communication, as in network protocols.

3 Listing of Publications

Peer Reviewed Journals

1. A. Finkel, S. P. Iyer and G. Sutre. Well-abstracted Transition Systems: Application to FIFO Automata. Accepted to appear in *Information and Computation.*

2. M. Narasimha, R. Cleaveland and S. Purushothaman Iyer. Role of observations in Probabilistic Open Systems. *Electronic Notes in Theoretical Computer Science*, Volume 27, 1999.

Refereed Workshops and Conferences

3. Dan DuVarney, Rance Cleaveland and Purush Iyer. A `C` interface to the Concurrency Workbench. In Proc of *Monterey Workshop on Engineering Automation for Software Intensive System Integration.* (ed) Luqi and Manfred Broy. June 2001.

4. Parosh Abdulla, Christel Baier, S. Purushothaman Iyer and Bengt Jonsson. Reasoning about probabilistic lossy channel systems. In *CONCUR – International Conference on Concurrency Theory – 2000.* Lecture Notes in Computer Science, August 2000, PA.

5. Alain Finkel, S. Purushothaman Iyer and Gregoire Sutre. Well-Abstracted Transition Systems. In *CONCUR – International Conference on Concurrency Theory – 2000.* Lecture Notes in Computer Science, August 2000, PA.

6. Rance Cleaveland and S. Purushothaman Iyer. Branching-Time Probabilistic Model Checking. In *Proc. of the 8th Int. Workshop on Process Algebra and Performance Modeling (PAPM 2000)*, Carleton Scientific, Geneva (Switzerland), 15 July 2000.

7. Parosh Abdulla, S. Purushothaman Iyer and Aletta Nylén. Unfoldings of Unbounded Petri Nets. In Proceedings of *Computer-Aided Verification* '00. Lecture Notes in Computer Science, Chicago.

8. Parosh Abdulla, S. Purushothaman Iyer and Aletta Nylén. SAT-solving the coverability problem for unbounded petri nets. In Proceedings of *NSF/ARO Workshop on Software Systems.* (ed.) Egidio Astesiano. June 2000.

9. Purush Iyer. Branching-Time Logic and Semantics for Probabilistic Open Systems. Proc. of *Nordic Workshop on Programming Theory.* Oct 1999.

10. Murali Narasimha, Rance Cleaveland and Purush Iyer. Probabilistic Model-Checking via Modal Mu-Calculus. *Foundations of Software Science and Computation Structures*, a **TAPSOFT** conference, Amsterdam, the Netherlands, 22-26 March 1999.

11. M. Narasimha, R. Cleaveland and S. P. Iyer. The role of observations in probabilistic open systems. *Workshop on Engineering for Complex Software.* Monterey, CA, Oct 1998.

Papers under consideration

12. S. Purushothaman Iyer, Rance Cleaveland, M. Narasimha. A Branching-time theory of Probabilistic Processes. Under consideration by *Theoretical Computer Science.*

13. P. Abdulla, C. Baier, S. P. Iyer, B. Jonsson. Undecidability of reasoning about probabilistic lossy channel systems. Under consideration by *Information and Computation.*

14. P. Abdulla, S. P. Iyer and A. Nylén. SAT-solving the coverability problem for Petri Nets. Under consideration by *Formal Methods in System Design.*

4 List of participating personnel

The following people participated in the project:

1. Dan DuVarney, Graduate Student, North Carolina State University, Raleigh, NC 27695-7534. *ABD.* Will submit and defend his PhD dissertation by November 2001.

2. Murali Narasimha, Graduate Student, North Carolina state University, Raleigh, NC 27695-7534. *Graduated* with a PhD 1999. *Currently* at Ericsson R&D, RTP, NC

3. Rance Cleaveland, Professor, Department of Computer Science, State University of New York, Stony Brook, NY.

4. S. Purushothaman Iyer, Associate Professor, North Carolina State University, Raleigh, NC 27695-7534.